

**ASM INSTITUTE OF BUSINESS MANAGEMENT & RESEARCH
[IBMR], PIMPRI CHINCHWAD - 411019**

**A
PROJECT
REPORT
ON**

THE PLANT'S WORLD

SUBMITTED BY:

- 1) **Mr. Atharva Bhoyarkar (21024).**
- 2) **Mr. Pratik Joshi (21082).**
- 3) **Mr. Shafi Ansari (21004).**
- 4) **Ms. Mansi Pasari (21137).**
- 5) **Mrs. Trupti Tambe (21188).**

Under The Guidance Of

Prof: Shreya Shenai

Submitted To

**Savitribai Phule Pune University
[SPPU]**



**As a partial fulfillment for the award of the degree of
MASTER'S OF COMPUTER APPLICATION
Semester: 1**

At



**ASM's Institute Of Computer Studies, Pimpri, Pune. (Affiliated to
University of Pune & Approved by AICTE)**

2021 - 2023

INDEX

Sr No	Topic
Chapter 1	Introduction (Pg. No: 3)
1.1	Existing System and Need for System (Pg. No: 4)
1.2	Scope of Work (Pg. No: 4)
1.3	Operating Environment – Hardware and Software (Pg. No: 7)
1.4	Detail Description of Technology Used (Pg. No: 7)
Chapter 2	Proposed System
2.1	Proposed System (Pg. No: 9)
2.2	Objectives of System (Pg. No: 11)
2.3	User Requirements (Pg. No: 12)
Chapter 3	Analysis & Design
3.1	Use Case Diagram (Pg. No: 17)
3.2	Class Diagram (Pg. No: 21)
3.3	Sequence Diagram (Pg. No: 22)
3.4	Activity Diagram (Pg. No: 25)
3.5	Deployment Diagram (Pg. No: 29)
3.6	User Interface Design (Screens etc.) (Pg. No: 30)
3.7	E-R Diagram (Pg. No: 37)
3.8	Data Flow Diagram (Pg. No: 40)
3.9	Context Diagram (Pg. No: 41)
3.10	Feasibility Study (Pg. No: 44)
Chapter 4	User Manual
4.1	User Manual (Pg. No: 45)
4.2	Operations Manual / Menu Explanation (Pg. No: 47)
4.3	Reports (Pg. No: 47)
4.4	Table Specifications (Pg. No: 48)
Chapter 5	Drawbacks And Limitations (Pg. No: 51)
Chapter 6	Proposed Enhancements (Pg. No: 52)
Chapter 7	Conclusion (Pg. No: 53)
Chapter 8	Bibliography (Pg. No: 53)
Chapter 9	Annexures
Chapter 10	Sample program code (Pg. No: 55)
Chapter 11	User Interface Screens (Pg. No: 66)
Chapter 12	Test Procedure (Pg. No: 74)

1: INTRODUCTION

Our project is about nursery management system. The nursery includes supply & maintenance of plants which helps our environment to become pollution free. Nursery mainly includes following section:-

- Plants**
- Lawn grass**
- Water Lilly's**
- Billing**

Plants need extensive care and maintenance for their good growth and yield which is provided by nursery. Though very few people know about nursery, now a day's nursery becoming important designation.

Due to software management all the important procedure such as watering, fertilization, cutting will get easier and accurate than that of manual. Also we provide plant library which provides information about plants, their maintenance, plantation etc. So it can be referred easily.

Plants play important role in keeping environment pollution free and for our good health.

Therefore, well managed nursery will give us more advantage in this purpose.

1.1: NEED FOR SYSTEM

Now our system will overcome this all drawbacks. It will reduce efforts required to manage all shipping records. All work can be done on just few clicks. The Only need is to fill given forms for retrieving required information. This system will provides facilities like add user records, Update Supplier records, delete records, search document records. The system helps to maintain Inventory details and documents which are required for importing goods.

1.2: SCOPE OF WORK

Provide the information related to the business. Create a new user and gives rights to user according to role.

1. Admin:

> Inventory Control:

Admin has total control on inventory of product before import product from vendor admin first check product inventory and import product as per requirement. He also check inventory when customer place an order for particular product.

> Shipment Tracking:

Shipment tracking facility is available for all (i.e. admin, supplier and customer) Changes and updating is permissible to admin only. Shipment tracking is possible after consignment send successfully.

> Delivery details of product:

Admin is responsible for maintaining product delivery details. Delivery details include product name, type, ship name, consignment no, date of loading, expected date of delivery etc.

> Report Generation:

Admin generate different reports and maintain details. Reports are send to customer which required to receive consignment.

> Fulfillment of Order:

Admin is responsible for fulfillment of customer order on time. Admin must check inventory of product which are ordered by customer and send consignment to customer on time.

> Maintain Registration Details:

Admin made registration for supplier and customer for security purpose. After registration user_id and password is provided to concern supplier and customer.

➤ **Maintain Supplier Details:**

There are regular supplier for company who supply product to company. Admin maintain all details of suppliers company.

➤ **Maintain Customer Details:**

Customer made order for purchase of product when customer made an order it is in pipeline state. All the details of customer along with its order is maintain by admin.

➤ **Order Confirmation Details:**

Customer made order for different type of wood when customer made an order it is in pipeline state, there are different state such as Booking, Confirmation, and Finalization.

➤ **Document Details:**

Details of Documents needed for Shipment Tracking are Maintain by Admin.

➤ **Container Management:**

Admin is responsible for fulfillment of customer order within specific time. The products are sending in container using ship. To proper loading of product all this container details are maintained by admin and send to customer for loading of container on port.

2. Supplier

➤ **Filling Quotation Details:**

Supplier fills quotation details for supply of goods. Supplier provides product description and minimum price.

➤ **Loading Delivery:**

Product Supplier makes arrangement for delivery of product. He makes arrangement of consignment and requirement of necessary documents to send consignment. Supplier must load product for delivery.

➤ **View Shipment Tracking:**

Supplier can track shipment of consignment. He can track at what time consignment will to destination and current state of consignment.

➤ **Container Management:**

To proper loading of product all this container details are maintained by Supplier and send to customer for loading of container on port.

➤ **Document Details:**

Details of Documents needed for Shipment Tracking are known to Supplier.

3. Customer:

➤ Placed an Order:

Customer placed an order for woods of 40ft and 20ft as per requirements. Customer has to pay partial amount at time of booking an order. After partial payment booked order enter into confirmation state.

➤ Payment of bill:

Customer made partial payment at the time of booked an order. The remaining amount of bill should be paid after consignment received.

➤ View Shipment Tracking:

Customer has facility to track their product consignment. He can track at what date consignment will to his port etc.

➤ Document Details:

Details of Documents needed for Shipment Tracking are known to Customer.

1.3: Operating Environment

✓ **Hardware:**

Processor: Intel Pentium IV Processor.

HDD: Min 4 GB.

RAM: Min 256 MB RAM.

Printer: HP Laser Jet.

✓ **Software:**

Operating system: Windows XP

Front End: JAVA (1.4 or higher)

Back End: Microsoft ACCESS.

1.4: Detail Description of Technology Used:

✓ **JAVA:**

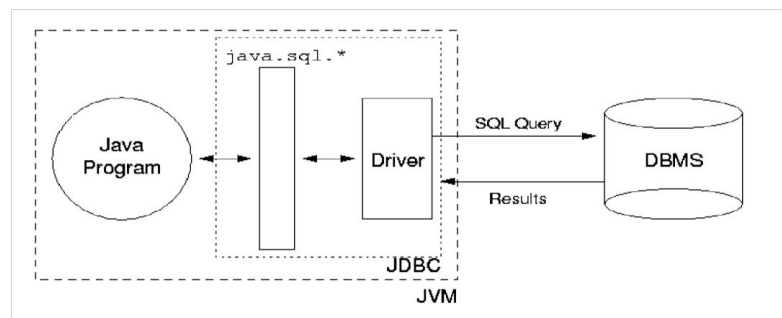
Java is a general-purpose, object-oriented programming language developed by Sun Microsystems of USA in 1991. Originally called Oak by James Gosling (one of the inventor of the language). Java was invented for the development of software for consumer electronic devices like TVs, toasters, etc. The main aim had to make java simple, portable and reliable.

Java Authors: James , Arthur Van , and others. Java is a high-level, third generation programming language, like C, FORTRAN, Smalltalk, Perl, and many others. You can use Java to write computer applications that play games, store data or do any of the thousands of other things computer software can do. Compared to other programming languages, Java is most similar to C. However although Java shares much of C's syntax, it is not C. Knowing how to program in C or, better yet, C++, will certainly help you to learn Java more quickly, but you don't need to know C to learn Java. A Java compiler won't compile C code, and most large C programs need to be changed substantially

before they can become Java programs. What's most special about Java in relation to other programming languages is that it

lets you write special programs called applets ,web project etc. that can be downloaded from the Internet and played safely within a web browser. Java language is called as an Object-Oriented Programming language and before beginning for Java, we have to learn the concept of OOPs(Object-Oriented Programming).

In the commercial world, we use Java 2 Enterprise Edition (J2EE) to solve business problems, to develop commercial software, or to provide contract services to other businesses' projects. If a company wants to build an e-business Website using a multitier architecture, it usually involves managers, architects, designers, programmers, testers, and database experts throughout the development lifecycle.



JDBC DRIVER MODEL

2: PROPOSED SYSTEM

2.1: Proposed System

- In the proposed system all the parameter are considered to maintain neat and easier solution.
- The company demanding quotation from different suppliers and select the quotation which provides maximum benefits at minimum price.
- Supplier first has to register himself to fill quotation details.
- Admin has to maintain Inventory details and documents which are required for importing goods.
- Admin is overall responsible for shipment documents, maintain supplier details, maintain customer details etc.
- Customer has to register himself to place an order.
- Customer makes payment of bill.
- The company is responsible for delivery or export of goods to customer.
- Supplier is responsible for Cost Calculation.
- Admin prepare reports.

2.2: Objectives of System

- It is very user-friendly and having added more features.
- To access more postal and parcel carriers.
- To improve our order and fulfillment operations.
- To integrate our customers.
- To develop global partnership.
- To keep track of the order according to its Status.

- To keep payment details, document details, container details etc.
- To issue tender Notice to suppliers.
- The System easily generates reports.
- Wastage of time is avoided.
- Provide security to data.
- Reduce manpower.
- Decrease manual mistakes.
- Easy maintenance of Import and Export document.

2.2: USER REQUIREMENTS

2.2.1: Functional Requirements

- In software engineering, a functional requirement defines a function of a software system or its component.
- A function is described as a set of inputs, the behavior, and outputs.
- Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describing all the cases where the system uses the functional requirements are captured in use cases.
- Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).

- Generally, functional requirements are expressed in the form "system must do

<requirement>", while non-functional requirements are "system shall be <requirement>". The plan for implementing *functional* requirements is detailed in the system *design*. The plan for implementing *non-functional* requirements is detailed in the system *architecture*.

- As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

2.2.1: Non Functional Requirements

➤ Product Requirements

- In the proposed system all the parameter are considered to maintain neat and easier solution.
- The company demanding quotation from different suppliers and select the quotation which provides maximum benefits at minimum price.
- Supplier first has to register himself to fill quotation details.
- Admin has to maintain Inventory details and documents which are required for importing goods.
- Admin is overall responsible for shipment documents, maintain supplier details, maintain customer details etc.

- **Customer has to register himself to place an order.**
- **Customer makes payment of bill.**
- **The company is responsible for delivery or export of goods to customer.**
- **Supplier is responsible for Cost Calculation.**
- **Admin prepare reports.**

2.2: Objectives of System

- **It is very user-friendly and having added more features.**
- **To improve our order and fulfillment operations.**
- **To integrate our customers.**
- **To keep track of the order according to its Status.**
- **To keep payment details, document details, container details etc.**
- **The System easily generates reports.**
- **Wastage of time is avoided.**
- **Provide security to data.**
- **Reduce manpower.**
- **Decrease manual mistakes.**

2.3: USER REQUIREMENTS

2.3.1: Functional Requirements

- In software engineering, a functional requirement defines a function of a software system or its component.
- A function is described as a set of inputs, the behavior, and outputs
- Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases
- Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).
- Generally, functional requirements are expressed in the form "system must do <requirement>", while non-functional requirements are "system shall be <requirement>". The plan for implementing *functional* requirements is detailed in the system *design*. The plan for implementing *non-functional* requirements is detailed in the system *architecture*.
- As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

2.3.2: Non Functional Requirements

➤ Product Requirements

❖ Usability requirements

Usability is the ease of use and learns ability of a human-made object. The object of use can be a software application, website, book, tool, machine, process, or anything a human interacts with. A usability study may be conducted as a primary job function by a usability analyst or as a secondary job function by designers, technical writers, marketing personnel, and others. Usability includes methods of measuring usability, such as needs analysis and the study of the principles behind an object's perceived efficiency or elegance. In human-computer interaction and computer science, usability studies the elegance and clarity with which the interaction with a computer program or a web site (web usability) is designed. Usability differs from user satisfaction and user experience because usability also considers usefulness.

❖ **Reliability requirements**

Reliability deals with the study, evaluation, and life-cycle management of reliability: the ability of a system or component to perform its required functions under stated conditions for a specified period of time. Reliability engineering is a sub-discipline within systems engineering. Reliability is theoretically defined as the probability of failure, the frequency of failures, or in terms of availability, a probability derived from reliability and maintainability. Maintainability and maintenance may be defined as a part of reliability engineering. Reliability plays a key role in cost-effectiveness of systems.

❖ **Portability requirements**

Portability in high-level computer programming is the usability of the same software in different environments. The pre requirement for portability is the generalized abstraction between the application logic and system interfaces. When software with the same functionality is produced for several computing platforms, portability is the key issue for development cost reduction.

- Transferring installed program files to another computer of basically the same architecture.
- Reinstalling a program from distribution files on another computer of basically the same architecture.

❖ Efficiency requirements

Resource consumption for given load describes efficiency of product and web site.

❖ Performance requirements

Performance metrics include availability, response time, channel capacity, latency, completion time, service time, bandwidth, throughput, relative efficiency, scalability, performance per watt, compression ratio, instruction path length and speed up.

- Short response time for a given piece of work
- High throughput (rate of processing work)
- Low utilization of computing resource(s)
- High availability of the computing system or application
- Fast (or highly compact) data compression and decompression
- High bandwidth / short data transmission time.

➤ Organizational Requirements

❖ Delivery requirements

Delivery requirements include details of delivery of product on time and as per client requirements. The products should be delivered on prescribed standard.

❖ **Implementation requirements**

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy.

an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium - recommended specifications, and software development tools contain implementations of programming languages.

❖ **Standard requirements**

The project should be developed as per standard format specified by IEEE.

Typical platforms include a computer architecture, operating system, programming languages and related user interface. The product should be developed as per client's standard requirements.

➤ **External Requirements**

❖ **Interoperability requirements**

Interoperability is a property of a product or system, whose interfaces are completely understood, to work with other products or systems, present or future, without any restricted access or implementation.

The IEEE Glossary defines interoperability as:

the ability of two or more systems or components to exchange information and to use the information that has been exchanged.

❖ Legislative requirements

In the proprietary software industry, an end-user license agreement or software license agreement is the contract between the licensor and purchaser, establishing the purchaser's right to use the software. The license may define ways under which the copy can be used. Software companies often make special agreements with large businesses and government entities that include support contracts and specially drafted warranties.

❖ Privacy requirements

The term "privacy" means many things in different contexts. Different people, cultures, and nations have a wide variety of expectations about how much privacy a person is entitled to or what constitutes an invasion of privacy. Privacy is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively. The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Privacy is sometimes related to anonymity, the wish to remain unnoticed or unidentified in the public realm.

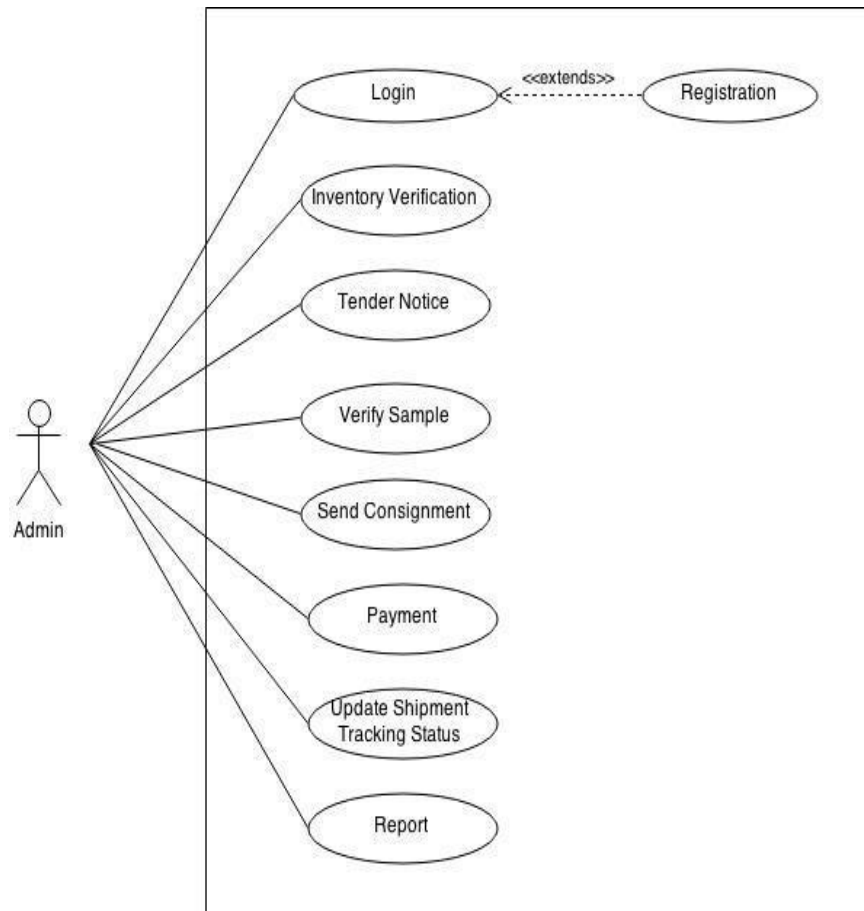
● Software Requirements Specification (SRS):

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional

and behavioural description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

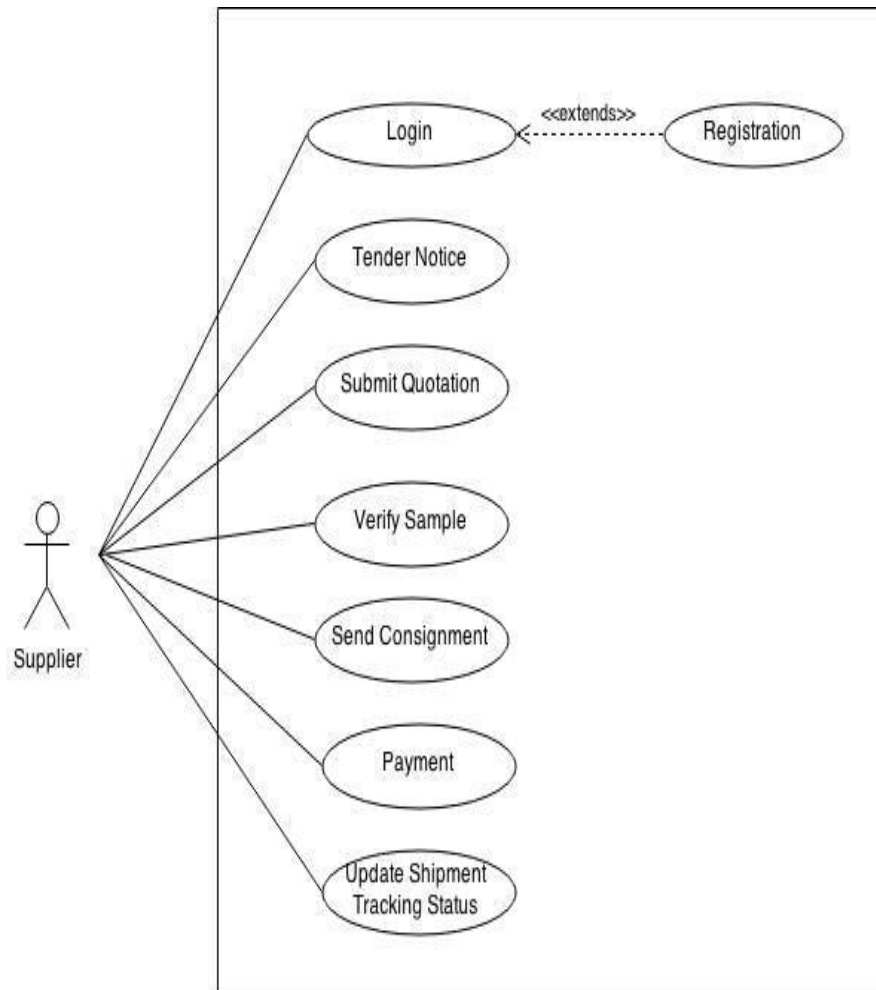
3: Analysis & Design

3.1: Use Case Diagram for Admin

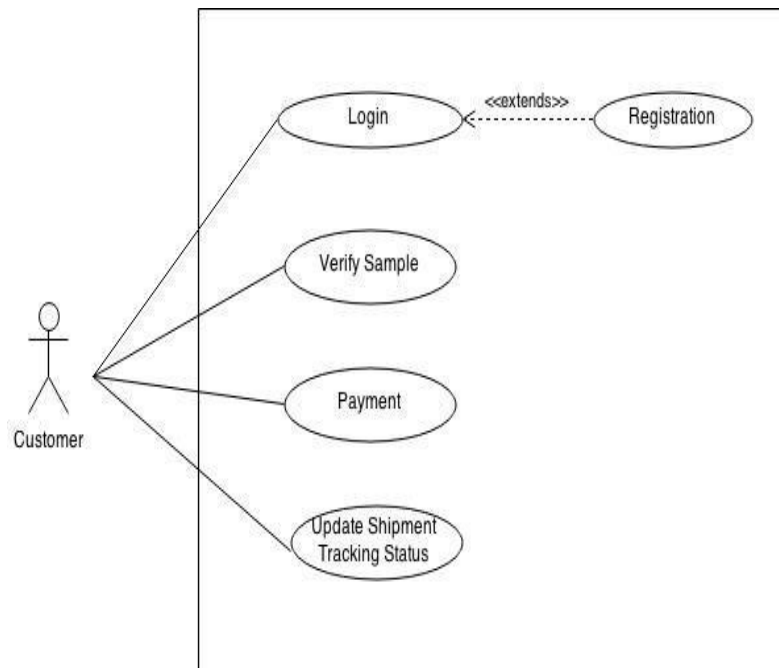


✓

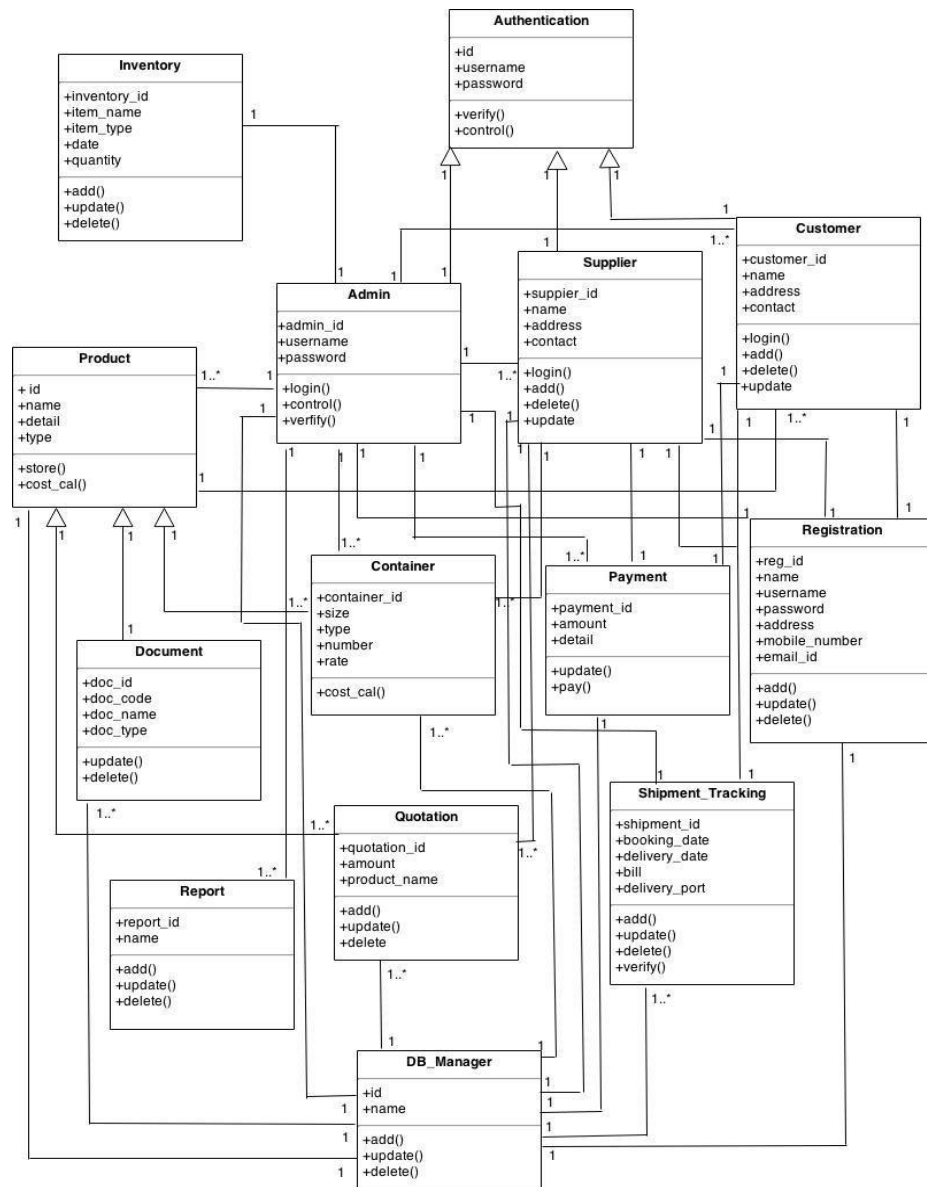
Use Case Diagram For Supplier



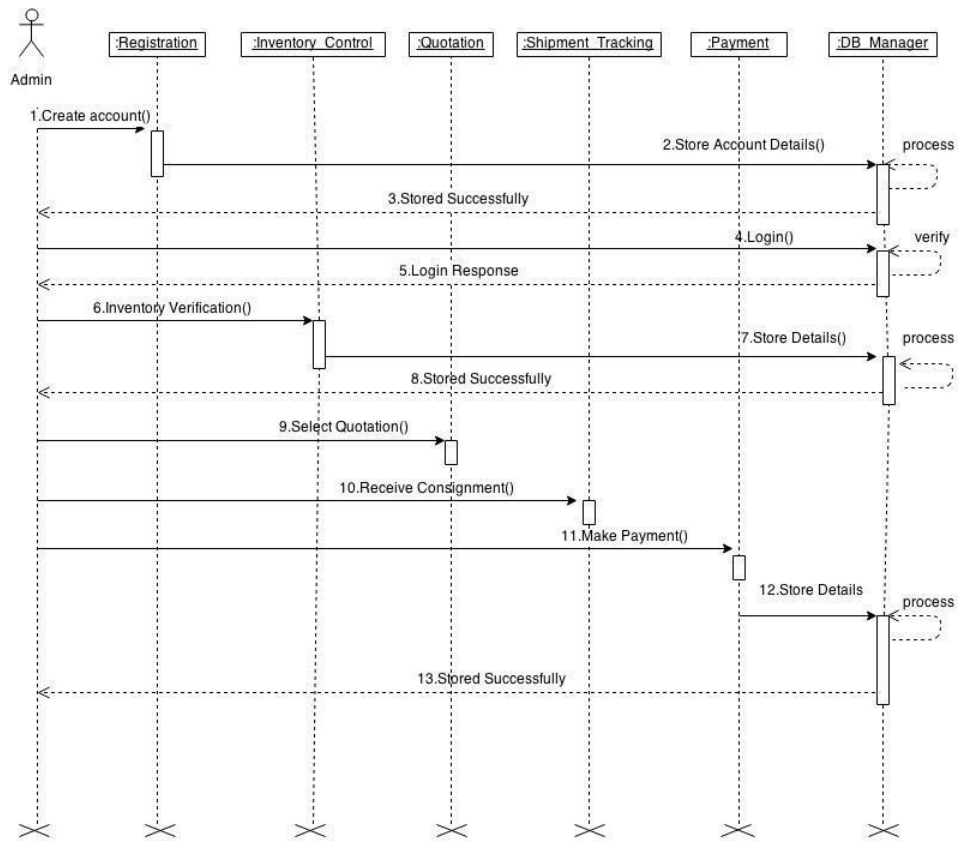
✓ **Use Case Diagram For Customer**



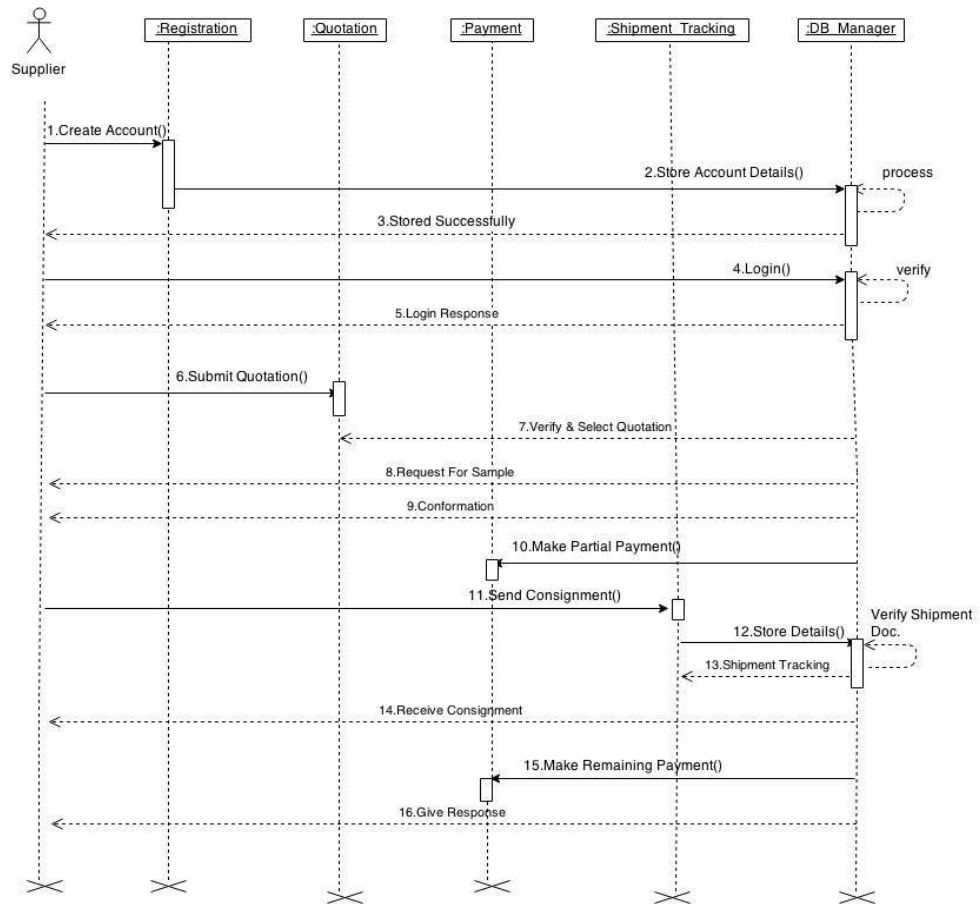
3.2: Class Diagram



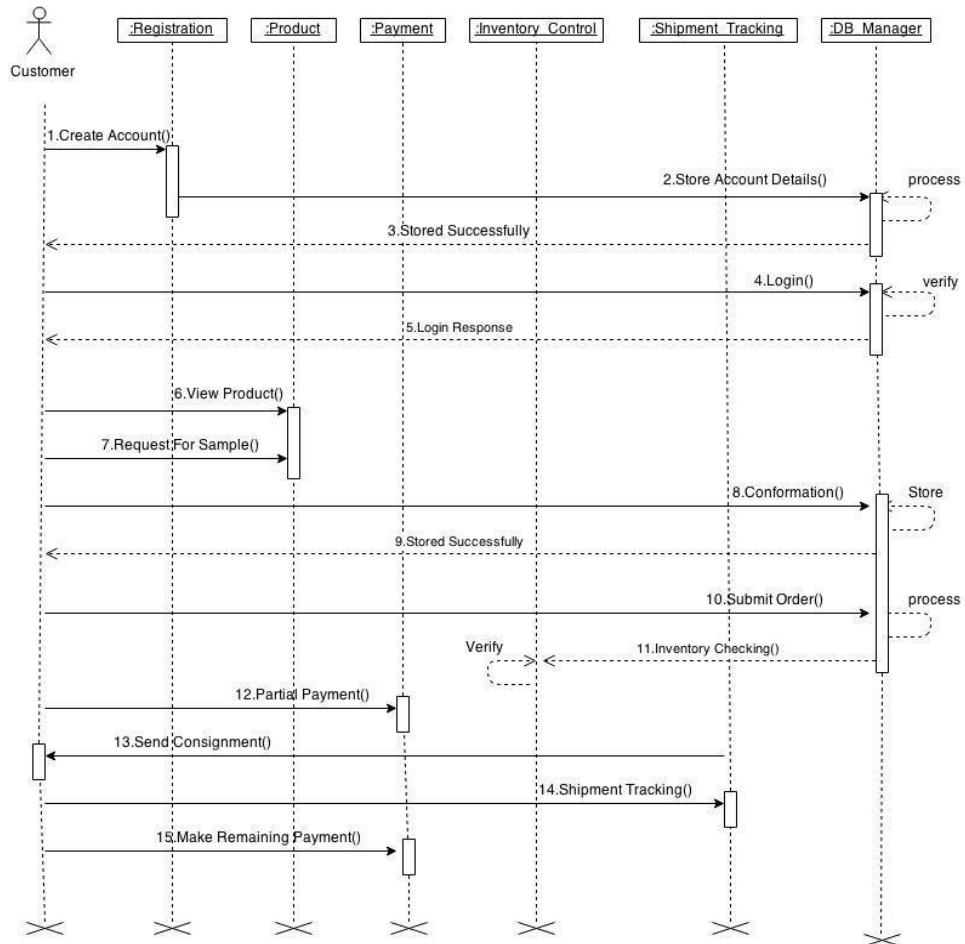
3.3: Sequence Diagram



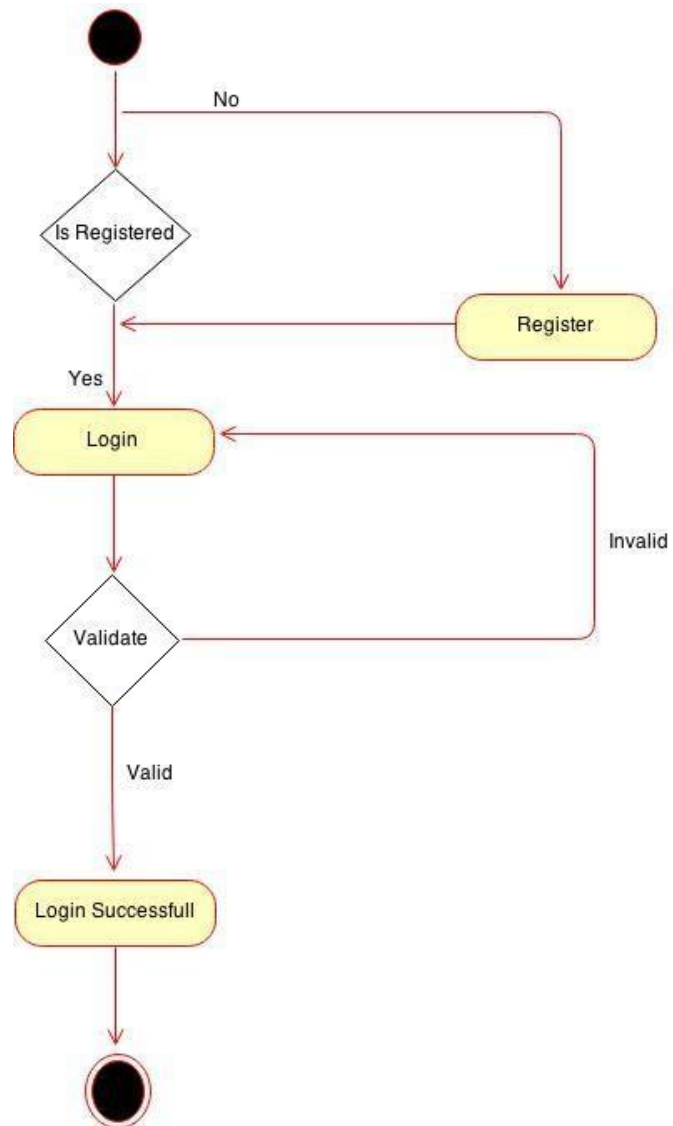
✓ **Sequence Diagram For Supplier**



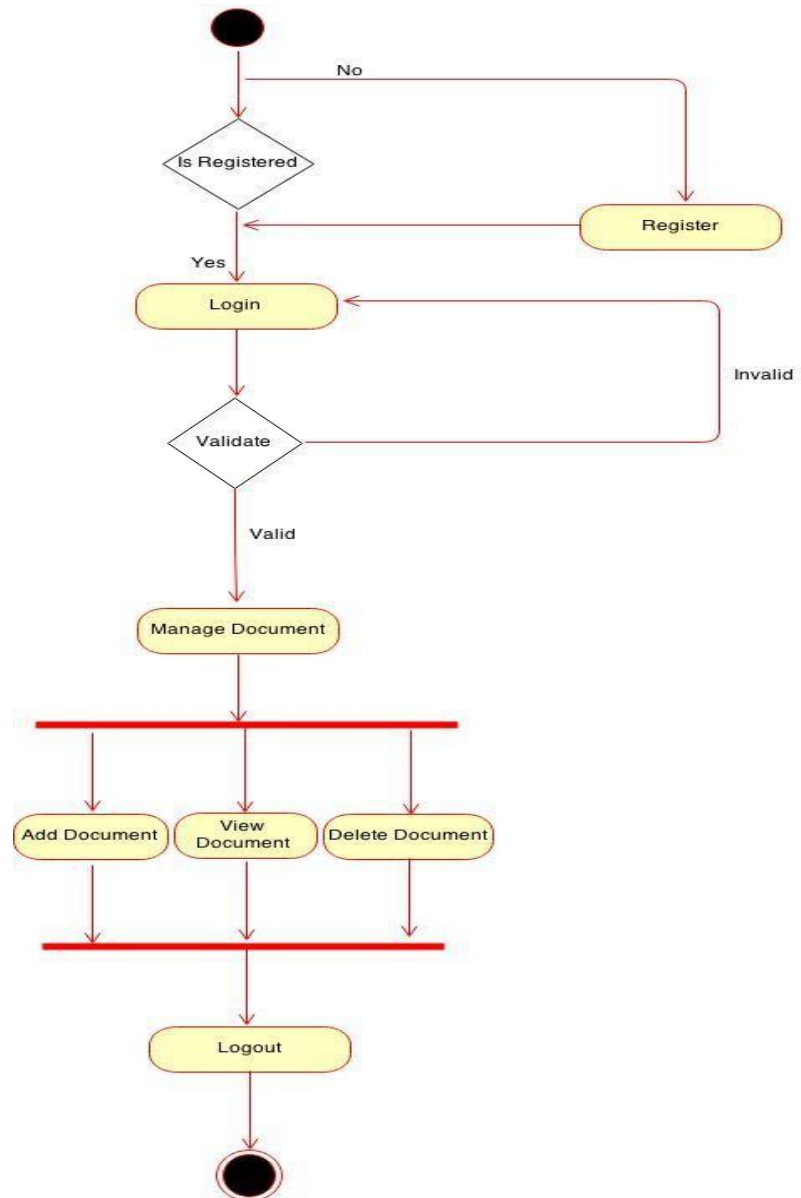
✓ **Sequence Diagram For Customer**



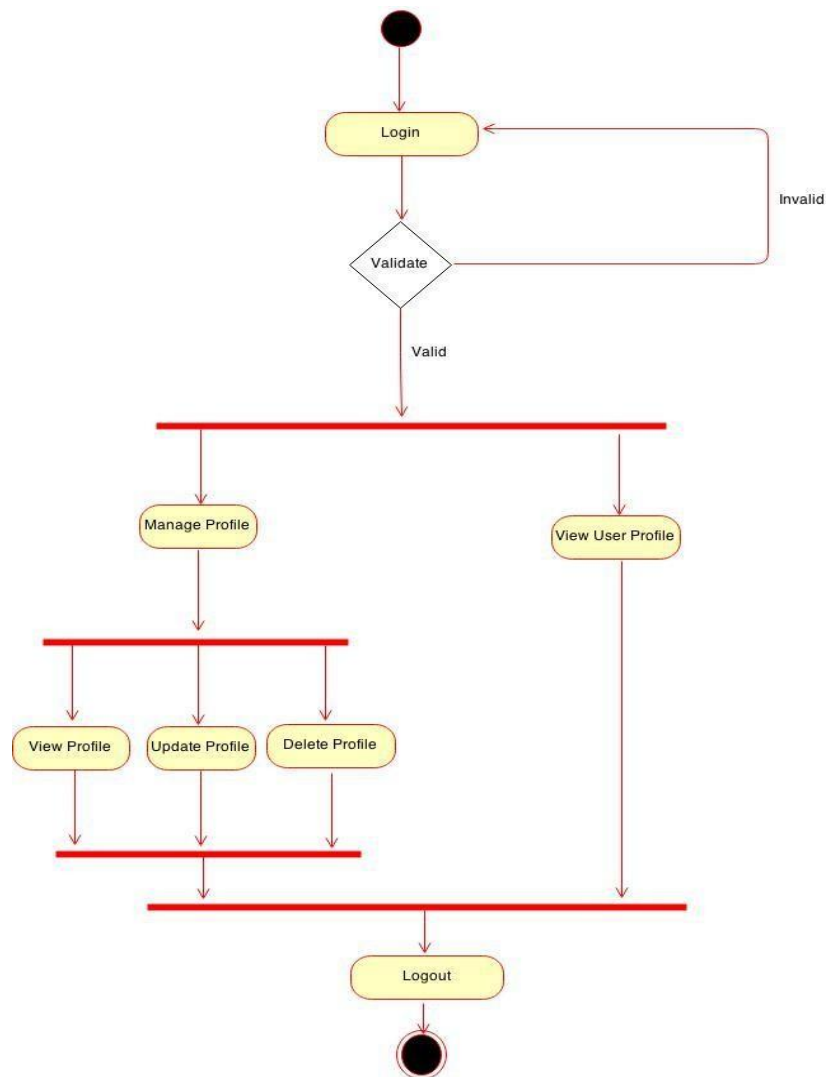
3.4: Activity Diagram For Registration



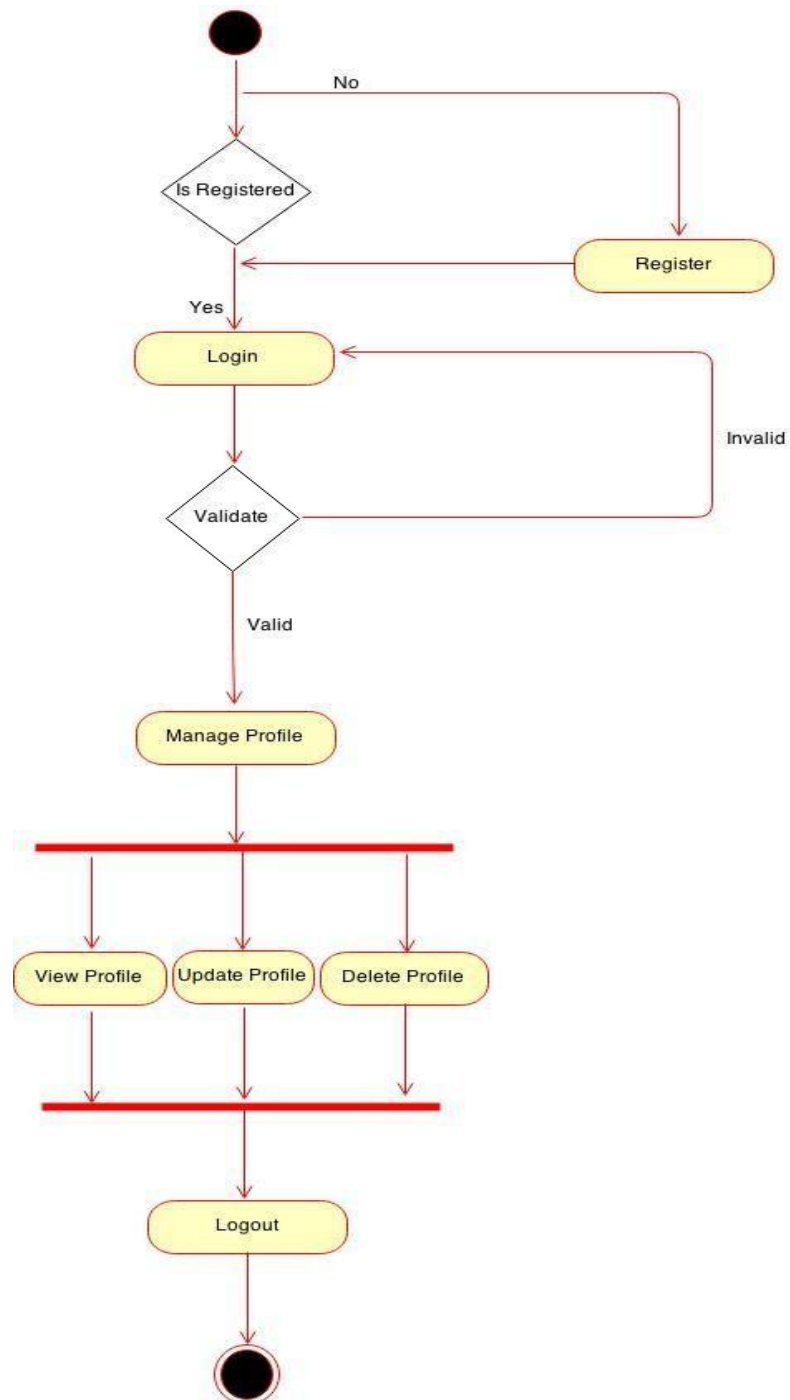
✓ Activity Diagram for Storing Document Details



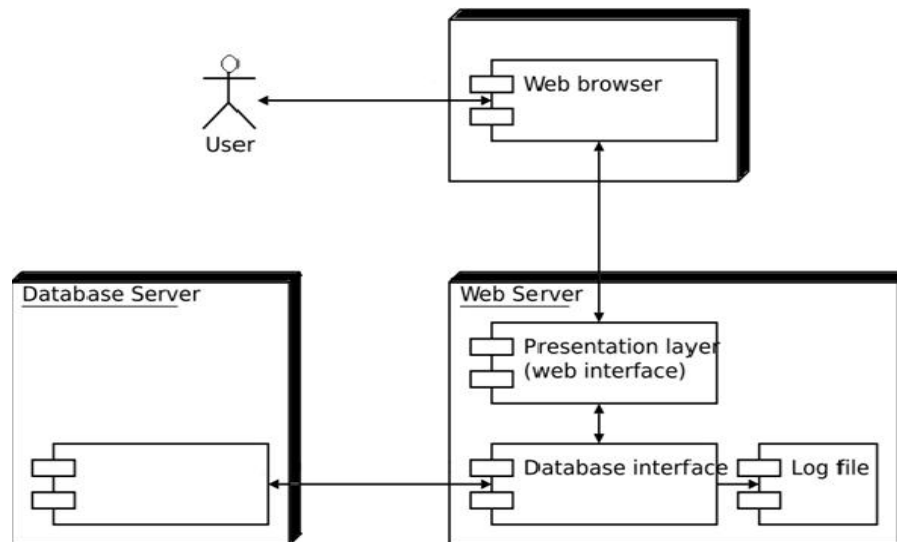
✓ Activity Diagram for Admin Profile



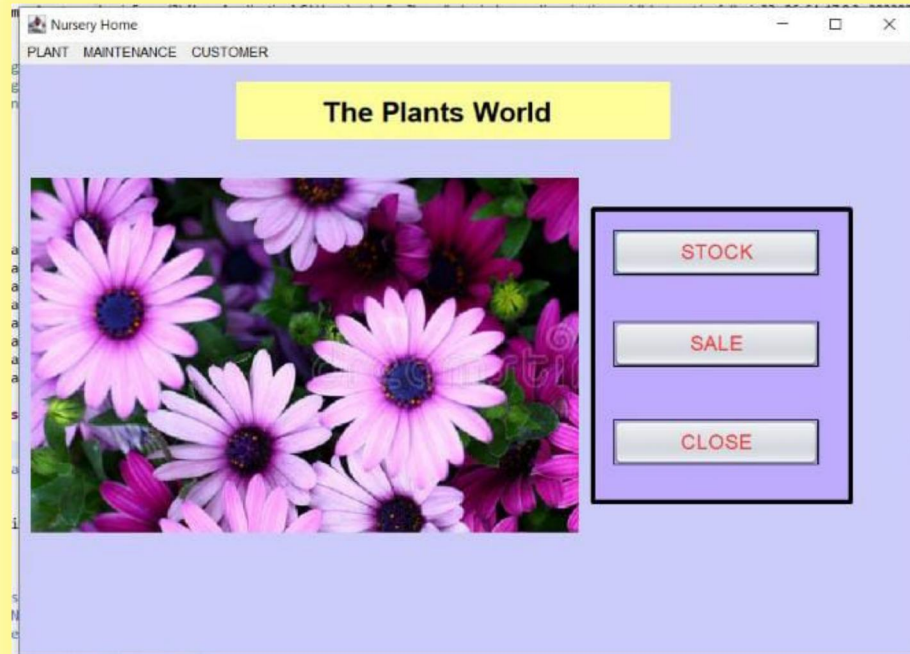
✓ Activity Diagram for User Profile



3.5: Deployment Diagram



3.6: User Interface Design (Screens etc.)



HOME

A login form window titled "Login Form" with a yellow glow. The window has a dark blue background and an orange header bar. The header bar contains the text "Login Form" and a close button "X". The form contains two input fields: "Username:" and "Password:". Below the input fields are two buttons: "Cancel" (red) and "Login" (blue). At the bottom of the form, there is a link that says "click here to create a new account".

LOGIN FORM

Register

- X

First Na...

Last Name:

Username:

Password:

Retype Pass:

BirthDate:

Address:

Cancel

Register

click here to login

REGISTER PAGE

The Plants World

CUSTOMER ISSUE

DATE

CUSTOMER NO:

CUSTOMER NAME:

CUSTOMER ADDRESS:

CUSTOMER ISSUE:

CUSTOMER ISSUE

CUSTOMER RECEIVE

The Plants World

CUSTOMER RECEIVE

DATE: 7/3/2022

CUSTOMER NO:

CUSTOMER NAME:

CUSTOMER ADDRESS:

Phone Number

SAVE CLEAR HOME

CUSTOMER RECIEVE

The image shows a web application window with a pink background. At the top, there is a title bar with a small icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area contains two labels, 'Plant Name' and 'Quantity', each followed by a text input field. Below these fields, there are three buttons arranged horizontally: 'Check Stock', 'Clear', and 'Home'. The entire window is highlighted with a yellow glow.

STOCK PAGE

The image shows a web application window titled "Add New Plants". The window has a light red background and a yellow glow effect. It contains the following elements:

- Title:** Add New Plants
- Form Fields:**
 - Plant Id:** A text input field.
 - Plant Category:** A text input field.
 - Plant Name:** A text input field.
 - Price:** A text input field.
 - Plant Quantity:** A text input field.
- Buttons:** Four buttons are located at the bottom of the form:
 - Search:** A button with a magnifying glass icon.
 - Add Plant:** A button with a plus icon.
 - Update:** A button with a refresh/circular arrow icon.
 - Home:** A button with a house icon.

ADDING NEW PLANTS (STOCK)

The Plants World

Plant Id Total Cost

Plant Name Payment

Plant Price Balance

Quantity

Id	Name	Price	Quantity	Total Cost
----	------	-------	----------	------------

BILLING PAGE

3.7: E-R Diagram

- **SYMBOL USED IN E-R DIAGRAM:**

The E-R model uses few basic concepts in producing an E-R diagram. These concepts are:-

i). Entity.

ii). Relationship.


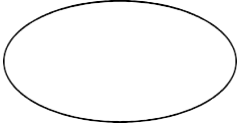
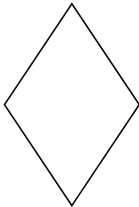

iii). Attribute.

1). **Entity**: An entity is an object or anything, which is distinguishable from objects.

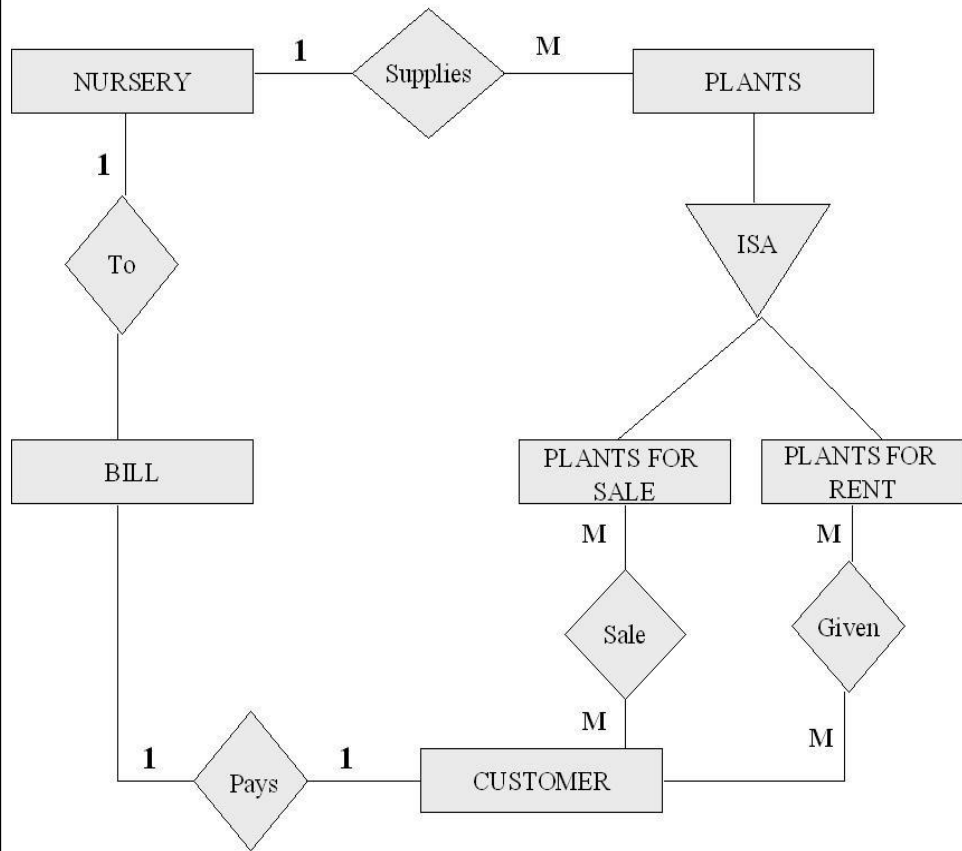
2). **Relationship**: A relationship is meaningful association, a linking or connection between entities.

3). **Attribute**: An attribute is any aspect quality or description of either an entity or relationship.

- **SYMBOL FOR E-R DIAGRAM:**

SYMBOL	SYMBOL NAME	REPRESENTS
	Rectangle	Entity Set
	Ellipse	Attribute
	Diamond	Relationship Set
	Line	Links Between Two Entity Set

E.R.D



E-R Diagram

3.8: DATA FLOW DIAGRAM

Data flow diagram is used to represent data & processes that manipulate it.

The data flow diagram enables the software engineer to develop the model of information domain & functional domain at same time. As the DFD is refined into greater levels of details, the analyst performs implicit functional decomposition of the system.

A data flow Diagram (DFD) is one of the popular graphical tool uses to depict the flow of data through a system. DFD shows the processes, data stores, data flow & the source & destination entries.

A few simple guidelines can aid immeasurably during derivation of data flow diagram.

1. The level 0 DFD should depict the system as a single bubble.
 2. The primary input & output should be carefully noted.
 3. Refinement should be by isolating candidate processes, data object & Stores to be represented at the next level.
 4. All arrows & bubbles should be labeled with meaningful names.
 5. Information flow continuity must be maintained from level to level.
- One bubble at a time should be refined.

✓ TYPES OF DFD's:

There are two types of DFD's as follows:-

- 1). Physical DFD's.
- 2). Logical DFD's.

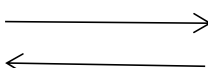
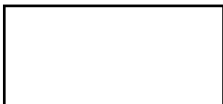
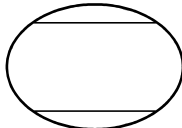

1). Physical DFD's: Physical DFD's depict the physical elements like people, report, documents, departments etc. Physical DFD's shows an implementation dependent view of the system.

2). Logical DFD's: Logical DFD's depict the logical elements like data process & events those are abstract than physical DFD's. Logical DFD's shows an implementation independent view of the system.

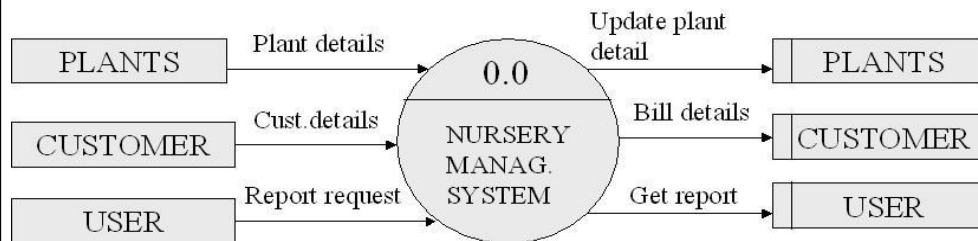
3.9: CONTEXT DIAGRAM

The top level diagram is called as a 'context diagram'. Context diagram contain single process, but it plays a very important role in studying the current system. Context diagram is constructed to show the highest level model of the system. This is the most general or broadcast picture of the current system. They are to represents the scope or boundaries of the system. Their purpose is identifying what is to include in the system under study.

1) . Symbol Used for Data Flow Diagram:

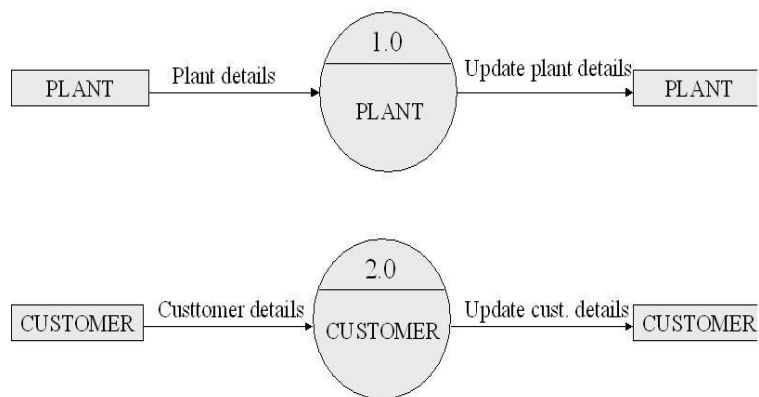
SYMBOL	SYMBOL NAME	REPRESENTS
	External Entity	A Source or destination of data which is external system.
	Data Flow	It is a packet of data. It may be in the form of document, letter, telephone call etc.
	Process	Here flow of data transferred
	Data Store	Any store data but with no difference to physical method of storing.

CONTEXT LEVEL DIAGRAM



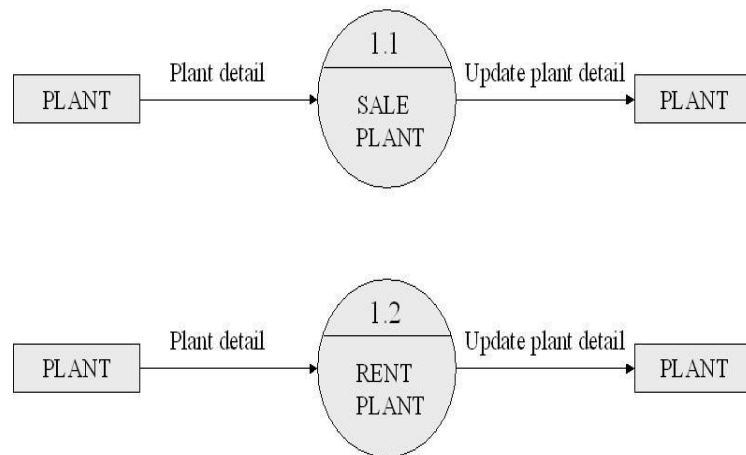
CONTEXT LEVEL DFD FOR PLANT NURSERY

FIRST LEVEL DFD



FIRST LEVEL DFD FOR PLANT NURSERY

SECOND LEVEL DFD



SECOND LEVEL DFD FOR PLANT NURSERY

3.10: FEASIBILITY STUDY

The feasibility of the system can be examined under. Technical feasibility, Economical feasibility & Operational feasibility.

- 1). **Technical Feasibility:** Technical feasibility plays an important role in feasibility study. The study reveals all the technical aspects & its corresponding results.
- 2). **Economical feasibility:** Economical feasibility is one of the most important aspects to be considered. This study reveals all the benefits & drawbacks in implementation of system. The total cost incurred for the development & implementation will be least as computer.
- 3). **Operational Feasibility:** Operational feasibility is the important part of feasibility study. We consider the capabilities of end user that how can easily handle the computer. In our projects as JAVA used which is GUI, due to which user can easily, handled it.

4: USER MANUAL

4.1 User Manual

Although the user interface of the system is constructed in such a way that anyone can use the system if he has the basic knowledge of the operating keyboard and mouse operation of the computer. All pages of the application contain the descriptive links and the buttons that will help the user to perform the required operation. There are following links/module.

1. Admin:

- Admin will have the full authority of the software.
- Admin will login by using his account.
- Admin will view/Edit the details.
- Admin will provide facility to give shipment tracking.
- Admin will have to make account of user or reporter and edit it.

2. Supplier:

- Supplier has to view the tender notice issued by admin.
- The supplier has to fill the tender details, Quotation details etc.
- Supplier Calculate Cost.

3. Customers:

- Customer has purchase the order, View shipment tracker.
- The customer can also do the payment of bill.

4.2: Operational Manual / Menu Explanation:

- **Login:**

This page is used to login the user in Shipment tracking system. For this user has to enter with the proper id, password and Select his/her Role then only user get Home page otherwise he will get error message as “Login fail...!!!”.

- **User Registration:**

When new user does not have account then through this page he can register himself to the system. The user has to just do one thing that fill all details on registration form. Also every entry should be unique so that the problem of duplication is avoided and there no user with same details in the database.

- **Change Password:**

When user wants to change his password then he is transferred to this page. Here he has to enter user id and has to enter old password then new password then again enter new password for confirmations then click “SAVE BUTTON” .then you get message “Password change successfully...”

- **User Registration Form:**

This Form contains the Registration details.

Step1: Click on Create User menu.

Step2: Enter all details as per the form fields.

Step3: You can see the user registration form in tab format i.e. Personal information, Mail ID.

Step4: Hear you can continuously fill all tabs then click on “Submit Button”

For Submit form..

Step5: Reset Button for clear all fields of form...

- **Loading Port:**

Loading port is used to so that admin can know from where the order is loaded from which port.

- **Delivery Port:**

Delivery port is helpful to know that where the order is supposed to be delivered.

- **Shipment Tracker:**

Shipment Tracker is the page useful for tracking the order of products and from which supplier the order is been placed. It also helps in keeping track from which port the order is loaded and which port the order is to be delivered. The admin can track or know the payment details of the order like advance payment, due payment, total amount of the order.

- **Document Details:**

Document page is provided so that all the required documents details can be maintained for the purchase of the order.

4.3: REPORTS

- **Generate Report:**

The “REPORT” contain information about the Shipment tracking , Supplier details , order details, container details etc.

4.4: Table Specifications

1. Admin

Entity Name	Entity type & Size	Constraints	Description
admin_id	Int(10)	Primary Key	Admin Id
Username	varchar(20)	Not Null	Username for login
Password	varchar(20)	Not Null	Password for login

2. Customer

Entity Name	Entity type & Size	Constraints	Description
cust_id	Int	Primary Key	Customer Id
cust_fname	varchar(20)	Not Null	First Name
cust_lname	varchar(20)	Not Null	Last Name
supplier_id	Int	Foreign Key	Supplier Id

3. Supplier

Entity Name	Entity type & Size	Constraints	Description
supplier_id	Int	Primary Key	Supplier Id
supplier_name	varchar(20)	Not Null	Supplier Name
Address	varchar(40)	Not Null	Supplier Address
City	Varchar(10)	Not Null	City Name
State	varchar(20)	Not Null	State Name
Country	varchar(20)	Not Null	Country Name

office_number	varchar(8)	Not Null	Office Number
mobile_number	varchar(10)	Not Null	Mobile Number
email_id	varchar(20)	Not Null	Email Id
bank_name	varchar(20)	Not Null	Bank Name
account_number	varchar(15)	Not Null	Account Number
Website	varchar(20)	Not Null	Website

4. Registration

Entity Name	Entity type & Size	Constraints	Description
reg_id	Int	Primary Key	Registration Id
Name	varchar(40)	Not Null	Name
Username	varchar(40)	Not Null	Username
Password	Varchar(10)	Not Null	Password
Address	varchar(60)	Not Null	Address
City	varchar(20)	Not Null	City Name
State	varchar(20)	Not Null	State Name
Country	varchar(20)	Not Null	Country Name
email_id	varchar(20)	Not Null	Email Id
mobile_number	varchar(10)	Not Null	Mobile Number

5. Product

Entity Name	Entity type & Size	Constraints	Description
product_id	int	Primary Key	Pdoduct Id
product_name	varchar(20)	Not Null	Product Name
product_detail	varchar(20)	Not Null	Product Detail
product_type	varchar(20)	Not Null	Product Type

6. Payment

Entity Name	Entity type & Size	Constraints	Description
payment_id	Int	Primary Key	Payment Id
Amount	Int	Not Null	Amount
payment_detail	varchar(20)	Not Null	Payment Details

4.6: LIMITATIONS & FUTURE ENHANCEMENTS OF THE SYSTEM

- **LIMITATIONS:**

The most important limitation of the existing system is its Manual system.

- **Drawbacks of the manual system:**

1. Time required for validations and updating is more.
2. Accuracy is less and Incomplete also.
3. Checking stock is difficult.
4. Repetition of work is going on.

- **FUTURE ENHANCEMENT:**

The computerized “purchase, sale & stock control system” is made with the intention to make easy to maintain the records and minimize the drawbacks of the manual System.

- **Advantages of computerized system over Manual system are:**

1. Computerized system is completely menu driven system, thus user can operate easily.
2. Time required is very less to make and search the records.
3. Computerized system generates various online records.
4. It reduces the data inconsistency and redundancy.
5. Computerized system is very helpful to display all the records.

4.7: Proposed Enhancements

Current system is designed in short amount of time so all functionality are not included in the system. More functionality can be included in the system in feature to help user of the system. There is no calculator for customers. SMS message alert will be including in this system to help the supplier and customer to know the details status about shipment.

4.8: Conclusion

This project work holds a very important place in my life because it has given me my first chance to get a look and feel of the environment. During development of this system module I learn new technologies, which would serve me in future. We have tried our level best to develop a system according to user requirement. The “Plant Nursey” system has been developed with due sincerity and diligence by following standard development practices. The system delivered functionality as required by the customer satisfaction. The system has proved for the organization popularity between its customer and owner. Also we can’t ignore the drawbacks and limitation of our system and in future we will make the enhancement on the system. Altogether it was a great experience, and we have learned a lot during system development.

4.9: BIBLIOGRAPHY

Before and at the time of developing the project following books are feared which gear us seem important guidelines for designing and developing the project and project reports.

4.10: REFERENCE BOOKS

- ✓ The complete Reference – Herbert Schildt, Patrick Naughton.**
- ✓ Java 6 Programming – Black Book – Kogent.**
- ✓ Core Java Vol.II – Advanced Features – Cay Harstmann, Gary Cornell.**
- ✓ Database System Concept– Sudarshan & Silberschath & Korth.**
- ✓ Software Engineering.**
- ✓ A Practitioner’s Approach–Roger .S. Pressman.**

4.11: WEBSITES

- www.google.co.in
- www.wikipedia.org
- www.grandanhill.com
- www.gardenplans.com
- www.mygardenplans.com
- www.gardenplansireland.com
- www.applegategardens.co.uk
- www.the-landscape-design-site.com
- www.greatlandscapingideas.com
- www.flower-gardening-made-easy.com

✓ **ANNEXURES:**

A1: Sample Code

```
package com.plants.archive;

import com.plants.DatabasePipeline.GlobbleConnection;

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.util.Calendar;
import javax.swing.border.*;
import java.sql.*;

public class CustomerBill extends JFrame implements ActionListener
{

    JLabel l, l1, l2, l3, l4, l5, dt;
    JButton b1, b2, b3;
    JPanel p1, p2;

    JTextField customerId, customerName, customerAddress, Amount,
    Date;

    public CustomerBill()
    {
        super(" CUSTOMER RECEIVE ");

        p1 = new JPanel();
        p2 = new JPanel();

        l = new JLabel();
        l1 = new JLabel();
        l2 = new JLabel();
        l3 = new JLabel();
        l4 = new JLabel();
        l5 = new JLabel();
        dt = new JLabel();

        customerId = new JTextField();
        customerName = new JTextField();
        customerAddress = new JTextField();
        Amount = new JTextField();
        Date = new JTextField();

        b1 = new JButton();
        b2 = new JButton();
        b3 = new JButton();
```

```
Container contentPane = getContentPane();
contentPane.setLayout(null);
```

```
    p1.setLayout(null);
```

```
    p1.setBorder(new LineBorder(new Color(153, 51, 0), 2, true));
    p1.setForeground(new Color(153, 51, 0));
    p1.setBackground(new Color(250, 250, 250));
```

```
    Font title = new Font("Arial", Font.BOLD, 25);
    l.setFont(title);
    //l.setFont(Font.BOLD);
    l.setForeground(Color.black);
```

```
    l.setText(" The Plants World ");
    l.setHorizontalAlignment(SwingConstants.CENTER);
    getContentPane().add(l);
    l.setBounds(260, 15, 250, 40);
```

```
l1.setFont(new Font("Arial", Font.BOLD, 20));
```

```
    l1.setForeground(Color.DARK_GRAY);
    l1.setText("CUSTOMER RECEIVE");
    l1.setMaximumSize(new Dimension(150, 25));
    l1.setMinimumSize(new Dimension(150, 25));
```

```
    p1.add(l1);
    l1.setBounds(190, 25, 260, 30);
    dt.setFont(new Font("Arial", 1, 13));
    dt.setText("DATE:");
```

```
    dt.setMaximumSize(new Dimension(105, 15));
    dt.setMinimumSize(new Dimension(105, 15));
    p1.add(dt);
    dt.setBounds(20, 75, 150, 25);
```

```
l2.setFont(new Font("Arial", 1, 13));
```

```
    l2.setText("CUSTOMER NO:");
    l2.setMaximumSize(new Dimension(105, 15));
    l2.setMinimumSize(new Dimension(105, 15));
```

```
    p1.add(l2);
    l2.setBounds(20, 130, 150, 25);
    customerId.setFont(new Font("Arial", 1, 13));
    customerId.addActionListener(this);
```

```
    p1.add(customerId);
    customerId.setBounds(200, 130, 300, 25);
```



```

l3.setFont(new Font("Arial", 1, 13));
l3.setText("CUSTOMER NAME:");
p1.add(l3);
l3.setBounds(20, 185, 150, 25);

l4.setFont(new Font("Arial", 1, 13));
l4.setText("CUSTOMER ADDRESS:");
l4.setMaximumSize(new Dimension(105, 15));
l4.setMinimumSize(new Dimension(105, 15));

p1.add(l4);
l4.setBounds(20, 240, 150, 25);
l5.setFont(new Font("Arial", Font.BOLD, 14));
l5.setText("Phone Number");

l5.setMaximumSize(new Dimension(105, 15));
l5.setMinimumSize(new Dimension(105, 15));
p1.add(l5);
l5.setBounds(20, 295, 150, 25);

customerName.setFont(new Font("Arial", 1, 13));
customerName.addActionListener(this);
p1.add(customerName);
customerName.setBounds(200, 185, 300, 25);

customerAddress.setFont(new Font("Arial", 1, 13));
customerAddress.addActionListener(this);
p1.add(customerAddress);
customerAddress.setBounds(200, 240, 300, 25);

Amount.setFont(new Font("Arial", 1, 13));
Amount.addActionListener(this);
p1.add(Amount);
Amount.setBounds(200, 295, 300, 25);

Date.setFont(new Font("Arial", 1, 13));
Date.addActionListener(this);
p1.add(Date);
Date.setBounds(200, 75, 300, 25);

getContentPane();
getContentPane().add(p1);
p1.setBounds(40, 60, 670, 390);

p2.setLayout(null);
p2.setBackground(new Color(250, 200, 255));
p2.setBorder(new LineBorder(new Color(153, 51, 0), 2, true));
b1.setFont(new Font("Arial", 1, 12));
b1.setText("SAVE");

```

```

        b1.setBorder(new LineBorder(new Color(0, 0, 0), 1, true));
        b1.addActionListener(this);
        p2.add(b1);
        b1.setBounds(70, 20, 80, 20);

        b2.setFont(new Font("Arial", 1, 12));
        b2.setText("CLEAR");
        b2.setBorder(new LineBorder(new Color(0, 0, 0), 1, true));
        b2.addActionListener(this);

        p2.add(b2);
        b2.setBounds(180, 20, 90, 20);
        b3.setFont(new Font("Arial", 1, 12));
        b3.setText("HOME");

        b3.setBorder(new LineBorder(new Color(0, 0, 0), 1, true));
        b3.addActionListener(this);
        p2.add(b3);
        b3.setBounds(300, 20, 100, 20);

        contentPane.add(p2);
        p2.setBounds(40, 455, 670, 60);
        Calendar cal = Calendar.getInstance();
        Date.setText((cal.get(Calendar.DATE)) + "/" +
(cal.get(Calendar.MONTH) + 1) + "/" + (cal.get(Calendar.YEAR)));

        Date.setEditable(false);
        setVisible(true);
        setSize(800, 600);
        setLocation(400,150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void actionPerformed(ActionEvent e)
    {

        if (e.getSource() == b1)
        {
            try
            {
                Class.forName("com.mysql.cj.jdbc.Driver");
                Connection c1 =
                DriverManager.getConnection("jdbc:mysql://localhost:3306/nms?useS
SL=false", "root", "root");

                Statement s1 =
                GlobbleConnection.connection.createStatement();

```

```
String ss1 = ("insert into customer_receive values('" +  
customerId.getText() + "','" + customerName.getText() + "','"  
+ customerAddress.getText() + "','" + Amount.getText() + "','" +  
Date.getText() + "')");  
s1.executeUpdate(ss1);
```

```
JOptionPane.showMessageDialog(null,"Records For  
"+customerName.getText()+" Saved Successfully");
```

```
customerId.setText("");  
customerName.setText("");  
customerAddress.setText("");  
Amount.setText("");
```

```
}  
catch (Exception ex)  
{  
System.out.println(ex);  
}  
}
```

```
if (e.getSource() == b2)  
{
```

```
customerId.setText("");  
customerName.setText("");  
customerAddress.setText("");
```

```
Amount.setText("");  
Date.setText("");
```

```
}
```

```
if (e.getSource() == b3)
```

```
{
```

```
dispose();  
new Home();
```

```
}
```

```
}
```

```
}
```

✓ **A1: Add Plant's Code**

```
package com.plants.archive;

import com.plants.DatabasePipeline.GlobbleConnection;

import java.awt.BorderLayout;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.lang.Object;
import java.awt.Font;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.JTextField;
import javax.swing.JComboBox;
import javax.swing.DefaultComboBoxModel;
import javax.swing.JButton;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Logger;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.awt.Color;

public class AddPlants extends JFrame
{

    private JPanel contentPane;
    private JTextField pltname;
    private JTextField pltqty;
    private JTextField pltprice;
    private JTextField pltId;
    private JTextField pltcategory;
```

```

Connection con;
    PreparedStatement pst;
    ResultSet rs;
    public void Connect()
    {
try {

        Class.forName("com.mysql.jdbc.Driver");
con =
DriverManager.getConnection("jdbc:mysql://localhost/nms","root","r
oot");
        }
catch (ClassNotFoundException ex)
{
        System.out.println(ex);

        }
catch (SQLException e)
{
        e.printStackTrace();
        }

}
public AddPlants() {

        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 892, 539);
        contentPane = new JPanel();
        contentPane.setBackground(Color.PINK);
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

JLabel titleLabel = new JLabel("Add New Plants");

        titleLabel.setHorizontalAlignment(SwingConstants.CENTER);
        titleLabel.setFont(new Font("Arial", Font.BOLD, 25));
        titleLabel.setBounds(193, 21, 217, 61);
        contentPane.add(titleLabel);

JLabel lblNewLabel = new JLabel("Plant Name");
lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel.setFont(new Font("Arial",lblNewLabel.getFont().getStyle()
Font.BOLD, lblNewLabel.getFont().getSize() + 11));
lblNewLabel.setBounds(52, 197, 133, 25);

```

```
contentPane.add(lblNewLabel);

pltname = new JTextField();
    pltname.setFont(new Font("Arial", Font.BOLD, 20));

    pltname.setHorizontalAlignment(SwingConstants.CENTER);
    pltname.setBounds(227, 191, 211, 36);
    contentPane.add(pltname);
    pltname.setColumns(10);

JLabel lblPlantCategory = new JLabel("Plant Category");
    lblPlantCategory.setFont(new Font("Arial", Font.BOLD, 21));
    lblPlantCategory.setHorizontalAlignment(SwingConstants.
    CENTER);
    lblPlantCategory.setBounds(449, 129, 165, 27);
    contentPane.add(lblPlantCategory);

JLabel lblPlantQuantity = new JLabel("Plant Quantity");
    lblPlantQuantity.setFont(new Font("Arial", Font.BOLD, 21));
    lblPlantQuantity.setHorizontalAlignment(SwingConstants.
    CENTER);
    lblPlantQuantity.setBounds(52, 271, 165, 32);
    contentPane.add(lblPlantQuantity);

pltqty = new JTextField();

    pltqty.setHorizontalAlignment(SwingConstants.CENTER);
    pltqty.setFont(new Font("Arial", Font.BOLD, 20));
    pltqty.setColumns(10);
    pltqty.setBounds(227, 269, 211, 36);
    contentPane.add(pltqty);

JLabel lblDate = new JLabel("Price");
    lblDate.setFont(new Font("Arial", Font.BOLD, 21));
    lblDate.setHorizontalAlignment(SwingConstants.LEFT);
    lblDate.setBounds(459, 197, 133, 25);
    contentPane.add(lblDate);

pltprice = new JTextField();

    pltprice.setHorizontalAlignment(SwingConstants.CENTER);
    pltprice.setFont(new Font("Arial", Font.BOLD, 20));
    pltprice.setColumns(10);
    pltprice.setBounds(622, 191, 211, 36);
    contentPane.add(pltprice);

    JButton Addplt = new JButton("Add Plant");
    Addplt.setFont(new Font("Arial", Font.BOLD, 21));
    Addplt.setBounds(225, 380, 144, 42);
```

```

contentPane.add(Addplt);

Addplt.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        try
        {
            Class.forName("com.mysql.cj.jdbc.Driver");

            Connection c1 =
            DriverManager.getConnection("jdbc:mysql://localhost:3306/nms?useS
            SL = false", "root", "root");

            Statement s1 = GlobbleConnection.connection.createStatement();
            String ss1 = ("insert into stock values('" + pltId.getText() + "','"
            + pltname.getText() + "','" + pltcategory.getText() + "','" +
            pltprice.getText() + "','" + pltqty.getText() + "')");

            s1.executeUpdate(ss1);

            JOptionPane.showMessageDialog(null,"Records For
            "+pltname.getText()+" Saved Successfully");

            pltId.setText("");

                                pltname.setText("");
                                pltcategory.setText("");
                                pltprice.setText("");
                                pltqty.setText("");
        }
        catch (Exception ex)
        {
            System.out.println(ex);
        }
    }
}

JLabel lblPlantId = new JLabel("Plant Id");
    lblPlantId.setFont(new Font("Arial", Font.BOLD, 21));
    lblPlantId.setHorizontalAlignment(SwingConstants.CENTER);
    lblPlantId.setBounds(33, 130, 133, 25);
    contentPane.add(lblPlantId);

pltId = new JTextField();
    pltId.setHorizontalAlignment(SwingConstants.CENTER);
    pltId.setFont(new Font("Arial", Font.BOLD, 20));

```

```

pltId.setColumns(10);
        pltId.setBounds(226, 124, 211, 36);
        contentPane.add(pltId);

pltcategory = new JTextField();

        pltcategory.setHorizontalAlignment(SwingConstants.CENTER);
        pltcategory.setFont(new Font("Arial", Font.BOLD, 20));
        pltcategory.setColumns(10);
        pltcategory.setBounds(622, 124, 211, 36);
        contentPane.add(pltcategory);

JButton btnNewButton = new JButton("Search");
        btnNewButton.setFont(new Font("Arial", Font.BOLD, 21));
        btnNewButton.setBounds(65, 380, 133, 42);
        contentPane.add(btnNewButton);

JButton btnUpdate = new JButton("Update");
        btnUpdate.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
            }
        }

JButton btnHome = new JButton("Home");
        btnHome.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e)
            {

                dispose();
                Home h=new Home();
                h.setVisible(true);
            }
        }

        btnHome.setFont(new Font("Arial", Font.BOLD, 21));
        btnHome.setBounds(582, 380, 144, 42);
        contentPane.add(btnHome);

        btnUpdate.addMouseListener(new MouseAdapter()
        {
            @Override
            public void mouseClicked(MouseEvent e) {
                if(pltId.getText().isEmpty() || pltname.getText().isEmpty() ||
                pltcategory.getText().isEmpty() || pltqty.getText().isEmpty() ||
                pltprice.getText().isEmpty())
                {

```



```

        }
        else
        {

try
{
//con =
DriverManager.getConnection("jdbc:mysql://localhost/nms","root","r
oot");

String UpdateQuery = "Update stock set p_qty= '"+ pltqty.getText()
+" ' "+" where id='"+pltId.getText();

Statement Add = GlobleConnection.connection.createStatement();
Add.executeUpdate(UpdateQuery);

JOptionPane.showMessageDialog(null,"Plant Quantity Updated
Successfully.");

        }

catch(SQLException e1)
{
e1.printStackTrace();

        }

        }

    }

    btnNewButton.addMouseListener(new MouseAdapter() {
        @Override
        public void mouseClicked(MouseEvent e) {
            if(e.getSource() == btnNewButton)
            {

try
{
String pid = pltId.getText();
pst = GlobleConnection.connection.prepareStatement("select * from
stock where id =?");

pst.setString(1, pid);
rs = pst.executeQuery();

if(rs.next()==false)
{

```

```
JOptionPane.showMessageDialog(null,"Plant Not found");
```

```
}
```

```
else
```

```
{
```

```
String pname = rs.getString("p_name");  
pltname.setText(pname.trim());
```

```
String pcategory = rs.getString("p_category");  
pltcategory.setText(pcategory.trim());
```

```
String pqty = rs.getString("p_qty");  
pltqty.setText(pqty.trim());
```

```
String pprice = rs.getString("p_price");  
pltprice.setText(pprice.trim());
```

```
}
```

```
}
```

```
catch (SQLException e1)
```

```
{
```

```
e1.printStackTrace();
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

A2: User Interface Screens

1. HOME PAGE



2. LOGIN FORM



The image shows a login form window with an orange header bar containing the title "Login Form" and a close button "X". The main area has a dark blue background. It features two input fields: "Username:" and "Password:". Below the password field are two buttons: a red "Cancel" button and a blue "Login" button. At the bottom, there is a link that says "click here to create a new account".

Username:

Password:

[click here to create a new account](#)

3. REGISTER PAGE

The image shows a registration form titled "Register" with a dark blue background and an orange header bar. The form contains the following fields and elements:

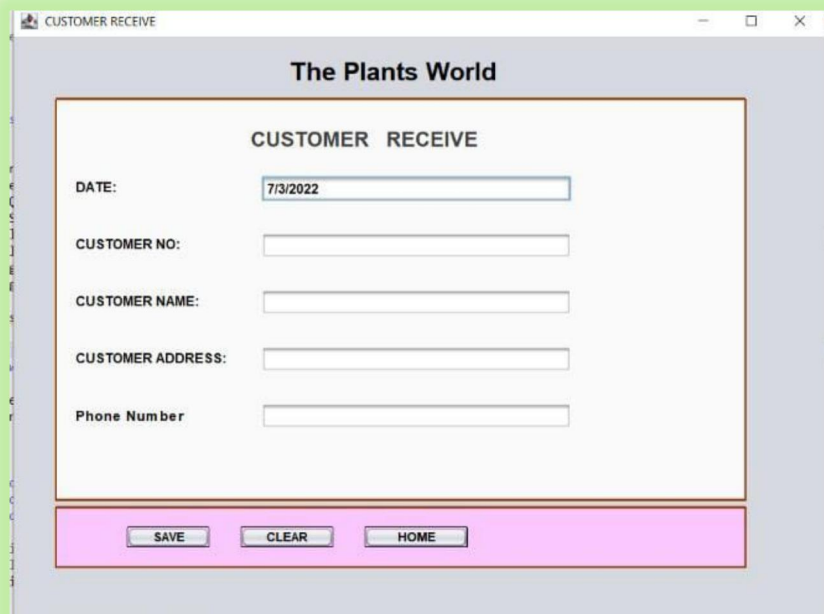
- First Na...:** A text input field.
- Last Name:** A text input field.
- Username:** A text input field.
- Password:** A text input field.
- Retype Pass:** A text input field.
- BirthDate:** A date input field with a calendar icon.
- Address:** A large text area for entering the address.
- Buttons:** A red "Cancel" button and a blue "Register" button.
- Link:** A link that says "click here to login" located below the buttons.

4. CUSTOMER ISSUE

The screenshot shows a web browser window with the title bar 'CUSTOMER ISSUE'. The main content area has a light blue background with a red border. At the top, it says 'The Plants World' in red. Below that, the title 'CUSTOMER ISSUE' is displayed in bold black text. The form contains five input fields with labels to their left: 'DATE' (containing '7/3/2022'), 'CUSTOMER NO:', 'CUSTOMER NAME:', 'CUSTOMER ADDRESS:', and 'CUSTOMER ISSUE:'. At the bottom of the form, there are three buttons: 'SAVE', 'CLEAR', and 'HOME'.

CUSTOMER ISSUE	
DATE	7/3/2022
CUSTOMER NO:	
CUSTOMER NAME:	
CUSTOMER ADDRESS:	
CUSTOMER ISSUE:	
<div>SAVE CLEAR HOME</div>	

5. CUSTOMER RECEIVE



The screenshot shows a software window titled "CUSTOMER RECEIVE" with a standard Windows title bar (minimize, maximize, close buttons). The main content area has a light gray background and is titled "The Plants World" in bold black text. Below this, there is a white rectangular box with a thin black border, also titled "CUSTOMER RECEIVE". Inside this box, there are five input fields with labels to their left: "DATE:" (containing "7/3/2022"), "CUSTOMER NO:", "CUSTOMER NAME:", "CUSTOMER ADDRESS:", and "Phone Number". Below the white box, there is a pink rectangular bar containing three buttons: "SAVE", "CLEAR", and "HOME".

CUSTOMER RECEIVE	
DATE:	7/3/2022
CUSTOMER NO:	
CUSTOMER NAME:	
CUSTOMER ADDRESS:	
Phone Number	

SAVE CLEAR HOME

6. STOCK PAGE



The image shows a web application window with a pink background. At the top, there is a title bar with a small icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area contains two input fields. The first is labeled "Plant Name" and is empty. The second is labeled "Quantity" and is also empty. Below these fields are three buttons: "Check Stock", "Clear", and "Home". The buttons are light gray with rounded corners and black text. The entire window is surrounded by a green glow effect.

Plant Name	<input type="text"/>
Quantity	<input type="text"/>
<div><button>Check Stock</button><button>Clear</button><button>Home</button></div>	

7. ADDING NEW PLANTS (STOCK)



Plant Name

Quantity

8. BILLING PAGE

The Plants World

Plant Id Total Cost

Plant Name Payment

Plant Price Balance

Quantity

Id	Name	Price	Quantity	Total Cost

5: Test Procedure

The software testing is the critical element of software quality assurance and represents the ultimate review of the software design and coding. The main objective of the testing is to find an error and to uncover the errors that are not yet discovered.

The increasing visibility of software as a system element and the attendant cost associated with a software failure and motivating forces for well planned, through testing. It is no unusual for a software development organization to expand between 30% to 40% of project effort on testing. In the extreme, testing of human related software can cost 3-5 time as much as all other software engineering activities combined. the testing phase involves the testing of the system using various test data, preparation of the test data plays a vital role in the system testing after preparing the test data, error was found and corrected by using the following the testing steps and correction are recorded for future reference. Thus, a series of testing is performed on the system before it is ready for implementation.

After completion of system analysis, d esign, and coding through testing of the system was carried out in a systematic approach, the main objectives of the system are:

- **To ensure that the operations of the system will perform as per the specification.**
- **To make sure that the system meets the user requirement during the operations.**
- **To cross check when correct input are filled into the system output are correct.**
- **To make sure that during the operation incorrect inputs and the outputs will be detected.**

In testing process the number of strategies h as been used as mentioned below,

- **Unit Testing.**
- **Integration Testing.**
- **Validation Testing.**
- **Black Box Testing.**
- **User acceptance Testing.**

➤ **Unit Testing:**

Unit testing focuses verification efforts on the smallest unit of the software design. Using the system test plan, prepare in the design phase of the system development as guide, important control path is tested to uncover error within boundary of the module. The interface of each of the module was tested to ensure proper flow of information into and out of the module under consideration. Each module will be tested individually to make the individual component error free. Also, other attached modules will be error free.

➤ **Integration Testing:**

Each module will be tested of its effect on other module by integrating the modules. This will remove further errors from the system and may also result in some changes in the individual module.

➤ **Validation Testing:**

At the culmination of the integration testing the software was completely assembled as package, interfaces have been uncovered, and a final series of software validation testing began. Here we test the system function manner that can be reasonably by the customer, the system was tested against system requirement specification.

➤ **Black Box Testing:**

After performing validation testing, the next phase is output test of the system since no system code is useful if it does not produce the desired output in desired format. By considering the format of the report/output, report/output is generated or displayed and tested.

➤ **User Acceptance Testing:**

User acceptance testing is used to determine the whether the software is fit for the user to use. The System under consideration was listed for user acceptance by keeping constant touch with the prospective user of the system at the time of design, development and making change whenever required.

